Отчёт по лабораторной работе №8  
Целочисленная арифметика многократной точности

Студент: Агеева Анастасия Сергеевна, 1032212304

Группа: НФИмд-02-21

Преподаватель: Кулябов Дмитрий Сергеевич,

д-р.ф.-м.н., проф.

Москва 2021

Содержание

# 1 Цель работы

Цель данной лабораторной работы изучение алгоритмов целочисленной арифметики многократной точности.

# 2 Задание

1. Реализовать программно алгоритм сложения неотрицательных чисел;
2. Реализовать программно алгоритм вычитания неотрицательных чисел;
3. Реализовать программно алгоритм умножения неотрицательных целых чисел столбиком;
4. Реализовать программно алгоритм умножения “быстрый столбик”;
5. Реализовать программно алгоритм деления многоразрядных целых чисел.

# 3 Теоретическое введение

**Длинная арифметика** [1] — выполняемые с помощью вычислительной машины арифметические операции (сложение, вычитание, умножение, деление, возведение в степень, элементарные функции) над числами, разрядность которых превышает длину машинного слова данной вычислительной машины. Эти операции реализуются не аппаратно, а программно, с использованием базовых аппаратных средств работы с числами меньших порядков. Частный случай — арифметика произвольной точности — относится к арифметике, в которой длина чисел ограничена только объёмом доступной памяти.

## 3.1 Применение

Длинная арифметика применяется в следующих областях:

* составление кода для процессоров (микроконтроллеров) низкой разрядности. Например, микроконтроллеры серии AVR имеют АЦП с разрядностью 10 бит и регистры с разрядностью 8 бит. Этого недостаточно для обработки информации с АЦП; без длинной арифметики не обойтись;
* **криптография**. Большинство систем подписывания и шифрования данных используют целочисленную арифметику по модулю m, где m — очень большое натуральное число, не обязательно простое. Например, при реализации метода шифрования RSA, криптосистемы Рабина или схемы Эль-Гамаля требуется обеспечить точность результатов умножения и возведения в степень порядка 10309;
* математическое (см. список ПО) и финансовое ПО. Результат вычисления на бумаге должен совпадать с результатом работы компьютера с точностью до последнего разряда. В частности, калькулятор Windows (начиная с Windows 95) проводит четыре арифметических действия с намного большей точностью, чем позволяет процессор x86. Для научных и инженерных расчётов длинная арифметика применяется редко, так как ошибки во входных данных обычно намного больше, чем ошибки округления;
* стандартная тема в спортивном программировании.

## 3.2 Необходимые аппаратные средства для работы с длинной арифметикой

Строго говоря, для реализации арифметики произвольной точности от процессора требуется лишь косвенная адресация; в арифметике фиксированной точности можно обойтись даже без неё. Тем не менее, определённые функции процессора ускоряют длинную арифметику, одновременно упрощая её программирование.

* Флаг переноса. Операции «сложить/вычесть с переносом», «циклический сдвиг через бит переноса».
* Косвенная адресация с автоинкрементом и автодекрементом (индексный регистр после операции увеличивается или уменьшается).
* Умножение (умножение слова на слово, результат — двойное слово), деление (с возможным переполнением).

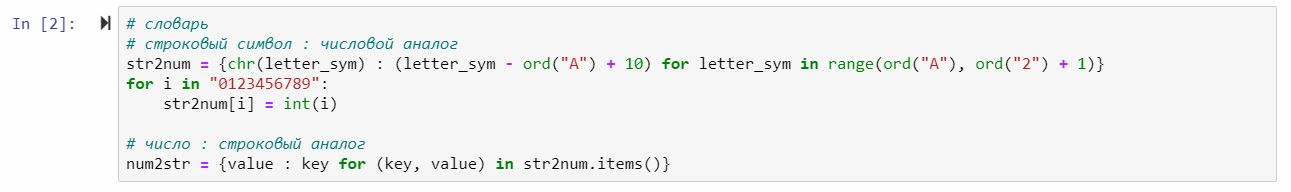
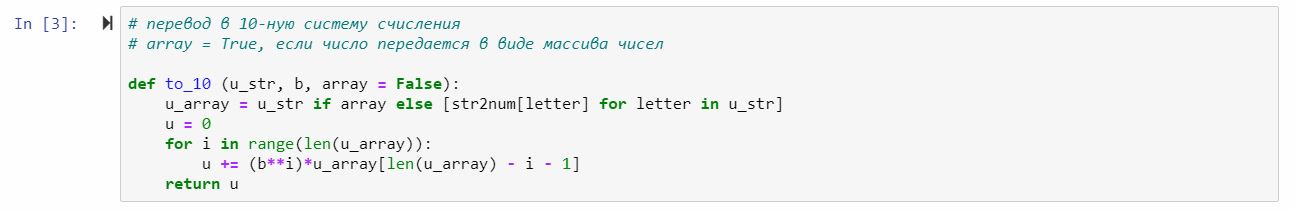
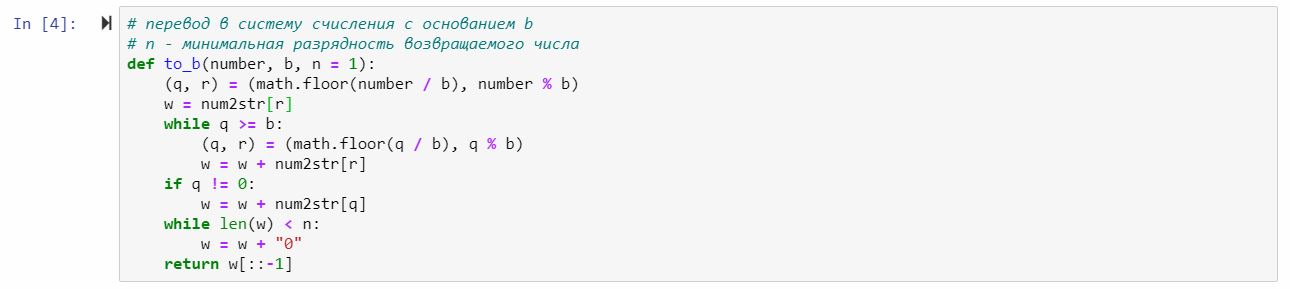
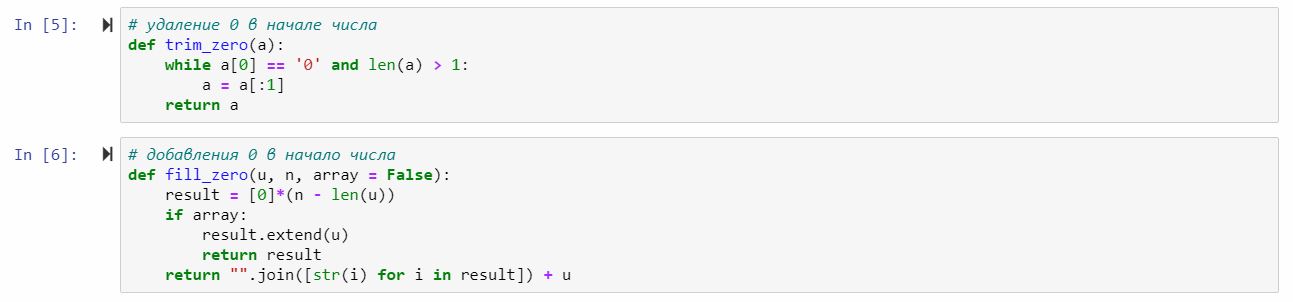
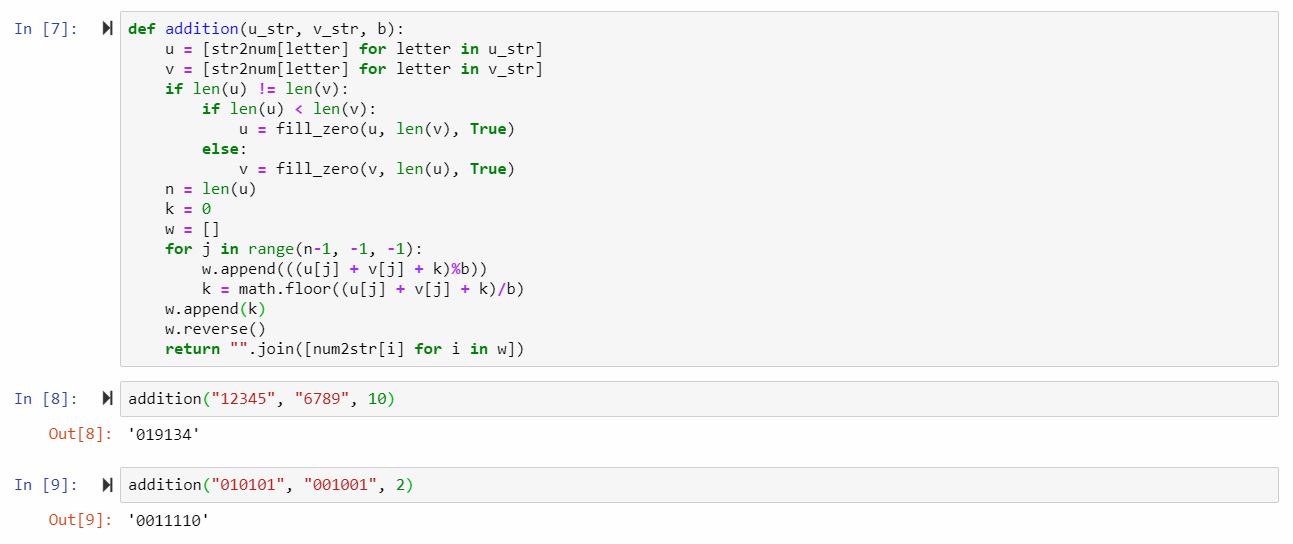
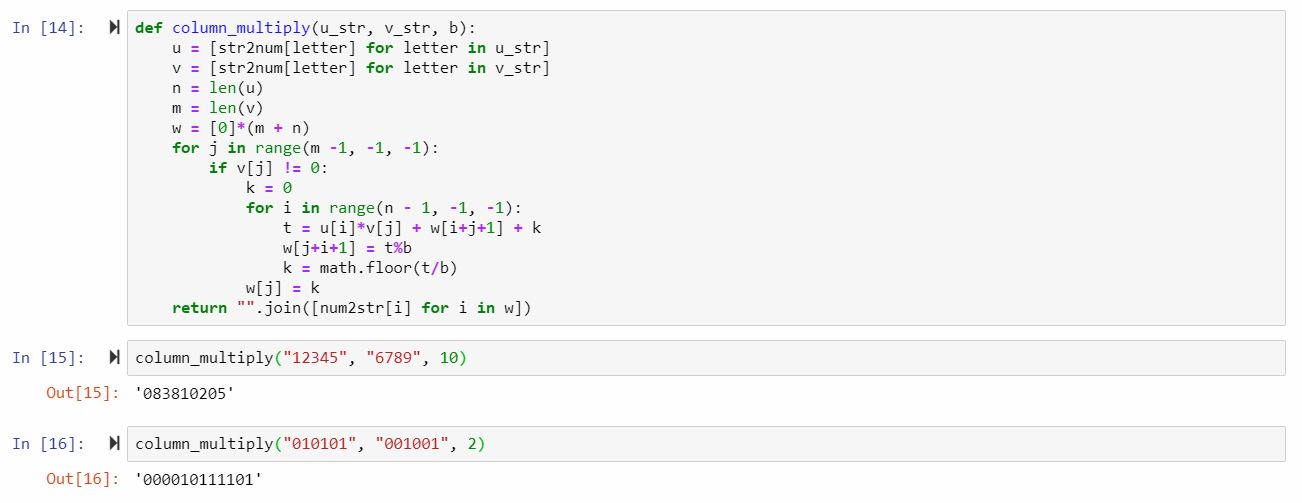
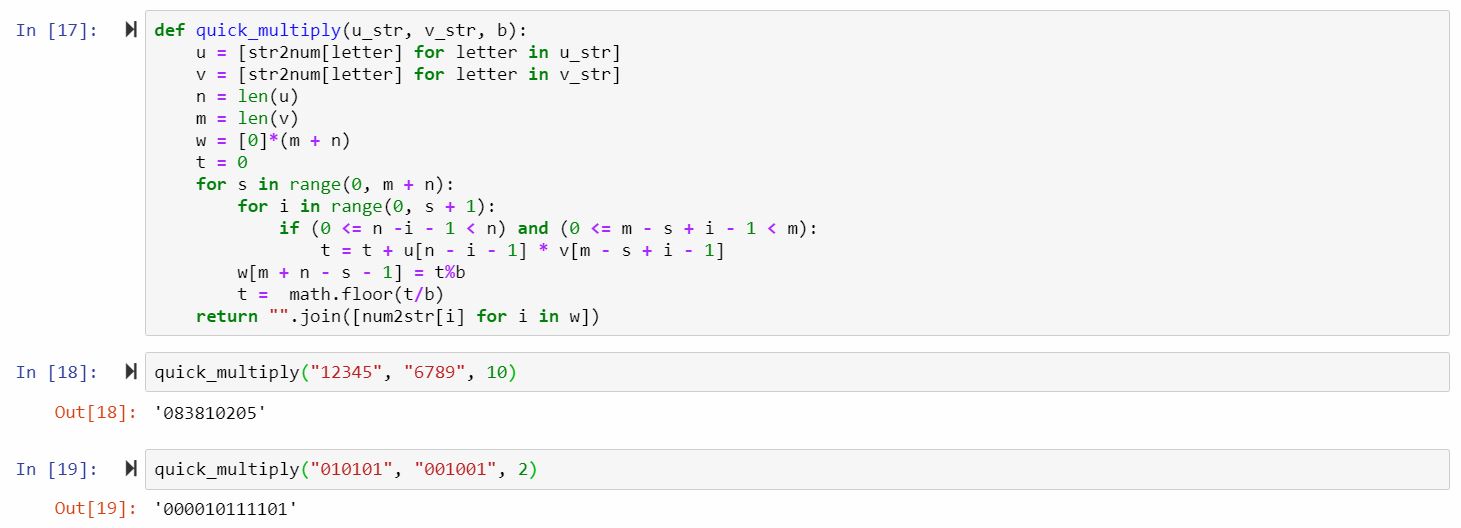
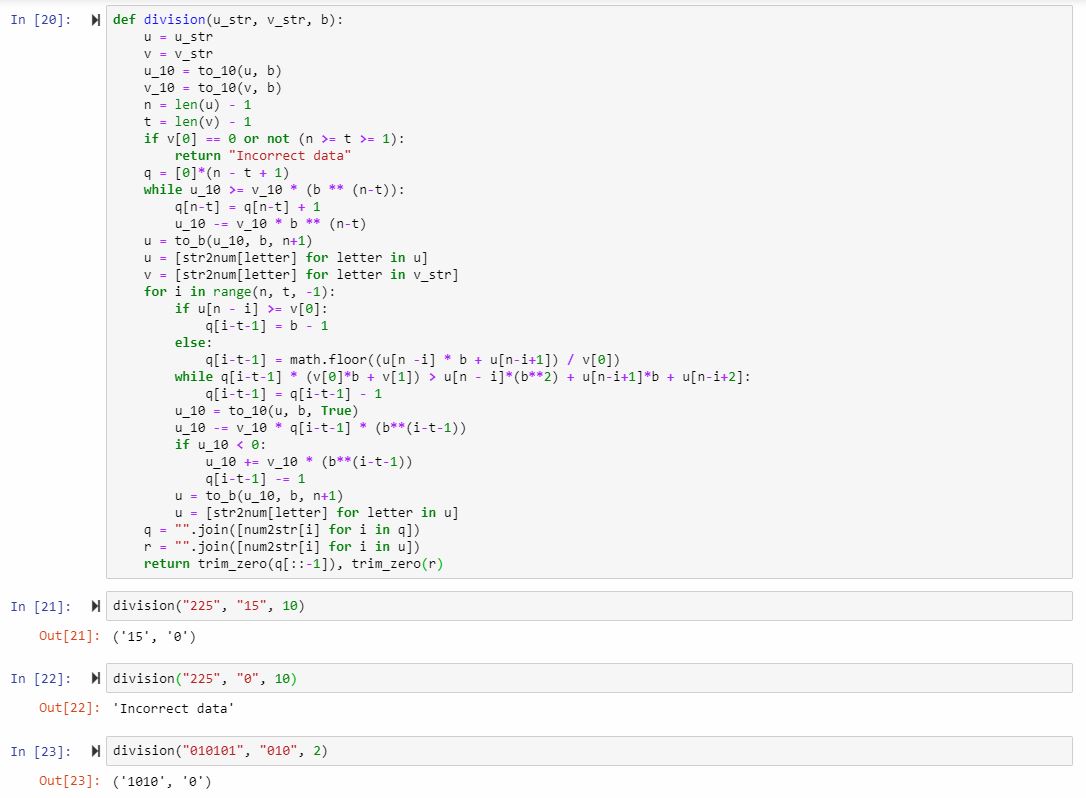
## 3.3 Реализация в языках программирования

Языки программирования имеют встроенные типы данных, размер которых, в основном, не превышает 64 бита (около 1019). Десятичная длинная арифметика была реализована в советских языках программирования АЛМИР-65 на ЭВМ МИР-1 и АНАЛИТИК на ЭВМ МИР-2. Для работы с большими числами, в современных языках программирования существует довольно много готовых оптимизированных библиотек для длинной арифметики.

Большинство функциональных языков позволяют переключаться с обычной арифметики на длинную без необходимости изменения кода арифметических расчётов. Например, Erlang и Scheme всегда представляют точные числа длинными. В Standard ML реализации всех разновидностей целых чисел определяются на основании сигнатуры INTEGER, позволяя выбирать необходимую размерность,— в том числе присутствует модуль IntInf, реализующий целые числа произвольной точности; в реализации PolyML этот модуль используется по умолчанию.

Встроенные библиотеки работы с большими числами есть в Ruby, Python и Java.

# 4 Выполнение лабораторной работы

1. Импортирую библиотеку math, которая понадобится для выполнения работы. Импорт библиотеки math
2. Задам два словаря, которые создают связь между числом и строковым символом (и обратно). 
3. Задам функцию для перевода числа в десятичную систему счисления. 
4. Задам функцию для перевода числа в систему счисления с основанием . 
5. Задам две функции, которые будут удалять или добавлять незначащие нули к числу. 
6. Задам функцию *addition()*, в которой реализовано сложение неотрицательных чисел по алгоритму, представленному в задании к работе. 
7. Задам функцию *substraction()*, в которой реализовано вычитание неотрицательных чисел по алгоритму, представленному в задании к работе. 
8. Задам функцию *column\_multiply()*, в которой реализовано умножение неотрицательных целых чисел столбиком по алгоритму, представленному в задании к работе. 
9. Задам функцию *quick\_multiply()*, в которой реализовано умножение неотрицательных чисел быстрым столбиком по алгоритму, представленному в задании к работе. 
10. Задам функцию *division()*, в которой реализовано деление многоразрядных целых чисел по алгоритму, представленному в задании к работе. 

# 5 Выводы

В ходе данной лабораторной работы я реализовала программно 5 алгоритмов целочисленной арифметики многократной точности.

# Список литературы

1. Длинная арифметика [Электронный ресурс]. Википедия, 2019. URL: <https://ru.wikipedia.org/wiki/Длинная_арифметика>.